Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUN 2007**	2. REPORT TYPE	3. DATES COVERED **00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE **Lean AISF:Applying COTS to System Integration Facilities**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **577 SMXS/FLT B,420 Richard Ray Blvd,Robins AFB,GA,31098-1640**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
CROSSTALK The Journal of Defense Software Engineering, June 2007

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**3**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Lean AISF: Applying COTS to System Integration Facilities

Harold Lowery
Warner Robins Air Logistics Center

Lean Avionics Integration Support Facility (AISF) is an initiative to introduce Lean concepts and methods to the F-15 AISF. Our strategy includes the use of commercial off-the-shelf (COTS) and open source software where appropriate. In this article, the author briefly describes the AISF and then discusses several examples of using COTS to reduce maintenance costs and improve performance.

Modern weapons are complex, high-performance systems. Much of the performance of a modern weapon system, as well as its complexity, derives from the software executing on computers embedded within it. It should come as no surprise that the engineering facilities used to develop and maintain these weapon systems are themselves complex systems that require considerable resources to operate and maintain. The application of *Lean* concepts enables significant cost reductions in the maintenance of system integration labs through the use of COTS items where appropriate. This article describes one such facility, the AISF located at Robins Air Force Base, and discusses several examples of how new technology impacts it.

Fighter AISF

In order to discuss Lean AISF, we first must discuss the Fighter AISF history.

History

The Fighter AISF is used to develop and maintain Operational Flight Program (OFP) software, primarily for the F-15 and other air combat platforms. The AISF achieved initial operation in the early 1980s and has been through several technology refresh cycles since then. The AISF includes a number of system integration benches. These benches are closed loop, hardware-in-the-loop systems consisting of avionics hardware, signal processing hardware and software, and simulation software. The OFPs execute in actual aircraft avionics with the airframe and operating environment simulated. The intent is that the OFP software cannot tell the difference between flying in an aircraft and flying in the lab.

Principles of Lean

In the early 1990s, researchers began discussing the concept of a *Lean* approach to manufacturing. Womack, Jones, and Ross introduced the term *Lean* when describing the Toyota Production System as part of a major study of the global automotive industry [1]. The concepts they described – focusing on the value stream and eliminating waste – have been successfully applied to manufacture and repair processes in the automotive and aerospace industries for some time. Innovative organizations are now applying *Lean* principles to their design and product development challenges. The emphasis in this domain is on eliminating waste, particularly in make-vs-buy decisions [2].

Application of Lean

Our initiative has the goal of transforming the traditional AISF to a Lean AISF, by moving from obsolete hardware/software to modern systems that are based on COTS equipment, open industry standards, and open source software where appropriate. In particular, we aim to lower the cost to support the AISF by applying Lean principles to product development to eliminate waste whenever possible. The expected benefits of this transformation are reduced hardware maintenance costs for AISF hardware, easier migration of new technology into existing AISFs, and reduced development costs for new AISFs to support weapon systems currently in development.

> *"The application of Lean concepts enables significant cost reductions in the maintenance of system integration labs through the use of COTS items where appropriate."*

Meeting the stringent real-time constraints of simulating a fighter requires significant computing horsepower. The first, second, and third generations of the AISF, like all system integration facilities built during the 1980s and 1990s, were based on expensive minicomputer hardware running proprietary operating systems and software development toolsets. In addition, a large investment in custom-designed hardware and software was necessary to meet the system's requirements, using the then-available technology. In implementing the fourth generation AISF, our aim is to eliminate waste, especially in the make-vs-buy decisions that so strongly drive life-cycle costs.

Example 1: Simulation Computers

For years, we have used minicomputers from a major simulation vendor to host our real-time simulation software. These machines have been true workhorses for us, but with the passage of time there were several reasons to move to newer technology. First, since these machines are based on the vendor's proprietary hardware, we have supported them via vendor maintenance contracts. This approach gave us superb support but was a strain on the budget. Second, as technology has advanced, our options for upgrading these computers grew limited. For example, the largest hard drives that they could accommodate are two gigabytes: This was great when the computers were new in 1991 but rather constrained some 15 years later.

We conducted a trade study to evaluate three alternatives. First, we could migrate from our existing simulation computers along the vendor's upgrade path to their next generation product. A significant feature of this alternative is the move to the open source Red Hat Linux operating system with the RedHawk real-time kernel. Second, we could build our own simulation hosts using COTS hardware running Linux. Third, we could expand the search space to the proprietary simulation products of other commercial vendors.

Alternatives one and two both used standard Intel-based servers running a version of Linux. The tremendous growth of the Internet had driven massive industry investment in servers, lowering the unit price of raw processing power. Alternative

one also included the vendor's proprietary hardware and software to provide a system optimized for real-time processing, albeit at a significantly higher price.

At first it was tempting to believe we could assemble a solution in-house by using off-the-shelf hardware (which we would buy strictly on price) and installing Linux. However, to re-host our legacy software to such a platform would require specific real-time capabilities – capabilities we would have to develop from scratch. As we began to tally up the engineering development costs, it became clear that cheap hardware could be too expensive.

Our trade study evaluated the alternatives on the basis of the following: 1) real-time capability, 2) supportability (over a nominal 10-year design life), 3) purchase costs, and 4) transition costs (including costs to re-engineer existing simulation software). Alternative one was the clear winner. The vendor's solution provided us an upgrade path where we could port our large legacy code base with minimal effort relative to other approaches. Although we could have bought equivalent hardware for half the price from other sources, the ability to quickly port our large legacy code base was a value proposition that surpassed the other alternatives.

Lesson Learned

Hardware may be cheap, but software engineers are expensive. When dealing with legacy systems, we found that the most cost-effective approach is generally the one that minimizes the software re-host effort.

Example 2: Bus Interface Cards

The H009 multiplex bus was an early forerunner of the Military Standard (MIL-STD)-1553B data bus that has become standard in military and even commercial aircraft. Since H009 was never as widely adopted as 1553B, there have always been relatively few suppliers of this hardware.

From the early days of the AISF, we made significant investments in designing, building, and maintaining custom H009 interface cards for the AISF. Our most recent implementation was designed in the early 1990s and had become unsupportable due both to obsolescence and personnel turnover. We had entered the H009 business simply because at the time we felt there were no viable commercial alternatives. In recent years, the engineering expertise to support this very specialized design across a small installed base (about 12 units, total) had eroded significantly. We had a strong desire to stop supporting in-house custom solutions, and by 2005 several vendors were offering H009 products.

As we investigated them, it quickly became clear that none would operate in our system without a significant rewrite of our existing software. In our third-generation hardware design, the software engineers had requested a number of features they thought would be needed. Over the last dozen years we had learned that some of those features were seldom, if ever, used – a form of waste. In effect, our board had been designed with some capabilities that were unnecessary and with others that were perhaps better done in software. In order to use the available COTS hardware, we would have to migrate some of the functionality of our custom hardware into an enhanced version of our software.

> *"Hardware may be cheap, but software engineers are expensive. When dealing with legacy systems, we found that the most cost-effective approach is generally the one that minimizes the software rehost effort."*

We had to trade off the costs of implementing a new custom hardware design and then supporting it for a number of years versus the one-time cost to modify the legacy interface software to accommodate the feature set offered by off-the-shelf solutions. Another factor we considered in our analysis was available support for the COTS equipment. Fortunately, the F-15 is gradually migrating away from the H009 bus to the much better supported MIL-STD-1553B bus. By provisioning the proper number of spare cards, we expect to support the H009 bus for as long as it remains in use on the aircraft.

Lesson Learned

A one-time investment of engineering dollars can be cost effective if it allows the use of COTS equipment and eliminates the engineering effort required to design and support an in-house solution over a period of years.

Example 3

In the first generation AISF, circa early 1980s, we used real aircraft control panels in our cockpit mock-ups. Although these gave the user a realistic experience in the lab, there were several drawbacks with them. Aircraft hardware is expensive to obtain, difficult to maintain, and has to be interfaced to the simulation computers using custom hardware. In our second-generation designs (late 1980s), we began experimenting with touch-screen equipped PCs as replacements for aircraft control panels. This approach eliminated aircraft hardware while still allowing us sufficient realism for the purposes of OFP development. However, implementing that approach required developing the software to display buttons, switches, etc., and to respond to the user as he or she activated these simulated controls. At the time, this meant a significant investment in custom software development.

Fast forward to 2006. Our original touch screen PC hardware had been replaced several times, but the software had been modified only slightly over the years and was in definite need of a major overhaul. But now we had options. The market for PC graphics software has greatly increased and several vendors offered promising products – the promise being that re-implementing our existing applications would be as easy as drawing the panels using the vendor's graphical editors. The old-timers among the technical staff were skeptical that it could be that easy, while the younger engineers were eager to try out new toys.

We evaluated various products and then made the investment. By using the vendor's tool, a trained engineer could prototype a control panel in a fraction of the time it would have taken with hand-crafted code.

However, what the tool saved us in creating panels it took back in time to integrate them when new hardware arrived. One significant problem involved a Linux driver that assumed a specific hardware configuration different from what we had purchased. In the end, a senior engineer rewrote the driver so that all the pieces would play together.

Lesson Learned

The young engineers were right that the COTS tools would simplify the process of generating control panels. But the grey-

beards were right too. There are always integration issues, and it is precisely this point where one vendor's product meets another's that the hard work usually takes place.

Conclusion

A Lean approach to AISF development and support would eliminate waste whenever possible. COTS products can be incorporated to great advantage if the engineering staff carefully weighs all alternatives when considering make-versus-buy decisions. In those cases where a COTS product is appropriate, it can eliminate the waste of supporting a custom solution using expensive in-house engineering talent. As always, it is important to clearly define the trade-offs and ramifications of using a COTS product.◆

References

1. Womack, James, Daniel Jones, and Daniel Ross. The Machine That Changed the World: The Story of Lean Production. New York: Rawson Associates, 1990.
2. Kearney, A.T. "The Line on Design – How to Reduce Material Cost by Eliminating Design Waste." Electronic Datasystems Corporation, 2003.

About the Author

Harold Lowery is the Director of B-Flight (F-15 AISF Support B227), 577th Software Maintenance Support Squadron. The majority of his 25 years of experience with the U.S. Air Force have been devoted to the development and support of F-15 avionics systems, with occasional excursions onto other Air Force platforms. When away from work, his five children consume the remainder of his time, energy, and patience.

577 SMXS/FLT B
420 Richard Ray BLVD
STE 100
Robins AFB, GA 31098-1640
Phone: (478) 926-9525
DSN: 468-9525
Fax: (478) 926-9525
DSN: 468-2682
E-mail: harold.lowery @robins.af.mil

Web Sites

Integration of Software-Intensive Systems (ISIS)

www.sei.cmu.edu/isis

The Software Engineering Institute ISIS initiative helps organizations successfully achieve system of systems interoperability by addressing the gaps between the net-centric vision and the status and capabilities of technologies that are being targeted to fulfill the vision, developing methods and techniques for helping organizations migrate to a network centric environment, identifying best practice for interoperability and integration, disseminating findings and guidance to the acquisition and development communities, maturing, and transitioning practical solutions into DoD organizations and the wider community.

Ten Commandments of COTS

https://acc.dau.mil/CommunityBrowser.aspx?id=24403

Interest in COTS products requires examination both in terms of its causes and effects, and in terms of its benefits and liabilities. The Defense Acquisition University offers some observations and voices some specific concerns and criticisms. They stress that their observations are essentially cautionary, not condemnatory: Huge growth in software costs will continue, not abate, and appropriate use of commercially available products is one of the remedies that might help to acquire needed capabilities in a cost-effective manner. Where use of an existing component is both possible and feasible, it is no longer acceptable for the government to specify, build, and maintain a comparable product.

COTS Journal Online

www.cotsjournalonline.com

Taking you into the world of the military acquisition machine, *COTS Journal* provides in depth coverage of commercially available embedded technology and its specific uses in military electronics and equipment. The subscription is free online, and archives can be accessed via the Web site without signing up just by clicking on the archives button on the menu bar.